FrameNet keretek és keretelemek felismerése neurális hálózatok és szódisztribúciós adatok felhasználásával

Tóth Ágoston

Debreceni Egyetem
Angol-Amerikai Intézet
Angol Nyelvészeti Tanszék
toth.agoston@arts.unideb.hu

Kivonat: Egyszerű visszacsatolt neurális hálózatok segítségével FrameNet-alapú keretszemantikai elemzést végeztem 9 különböző szóreprezentációs módszer felhasználásával 12 FrameNet keretre és ezek keretelemeire. A kipróbált szóreprezentációs eljárások között szerepeltek a szavak disztribúciós tulajdonságait leíró, nagy méretű korpuszból gyűjtött szóvektorok, melyek lehetővé tették a FrameNet keretek felismerését 91%-os pontossággal 86% fedés mellett (F-mérték: 89%), a keretelemek felismerése pedig 56%-os pontosságú volt 50%-os fedéssel (F-mérték: 53%). A disztribúciós szóábrázolások előnye az eltérő módszerekhez képest jelentős volt. A disztribúciós eszközök közül a környezetszavak leszámlálásán alapuló technikák és a neurális hálózatokban kialakuló prediktív szóbeágyazások egymáshoz hasonló teljesítményt nyújtottak ebben a kísérletben, a prediktív eljárások CBOW és SkipGram osztályai pedig közel azonos eredményt szolgáltattak.

1 Bevezetés

A jelentéselmélet három fő irányzata (strukturális, logikai és kognitív szemantika) kijelöli azokat a kereteket, amiben a jelentés gépi feldolgozásának a feladatait a számítógépes nyelvészet és a mesterséges intelligencia kutatásának vonatkozásában is elhelyezzük. Ebben a tanulmányban a kognitív nyelvészet számítógépes nyelvészeti szempontból kiemelt fontosságú eredményének, a FrameNetnek [1] a keretszemantikai kategóriáira támaszkodunk.

A mesterséges neurális hálózatokat gépi tanulási eszközökként egyre jobban megismerjük, napi gyakorisággal megtapasztaljuk széleskörű alkalmazási lehetőségeiket (többek közt az orvosi diagnosztika, arcfelismerés, tőzsdei árfolyamok megjósolása, időjárás-előrejelzés, gépi fordítás területén). A nyelvfeldolgozásban a szerepük túlmutat a más eszközökkel nehezen algoritmizálható részfeladatok végrehajtásán: a természetes nyelvek elsajátításának és feldolgozásának természetes közege az emberi neurális hálózat központja, az agy. A nyelvek kifejlődése, a mai nyelvek elsajátítása és használata is ehhez a közeghez kötődik.

A bemutatott kísérletsorozatban mesterséges neurális hálózatokkal FrameNet-alapú keretszemantikai elemzést végeztem megvizsgálva azt, hogy hogyan befolyásolta különböző szóreprezentációs módszerek használata a feladat megoldásának eredményességét. A kipróbált szóreprezentációs eljárások között szerepeltek a szavak diszt-

ribúciós tulajdonságait közvetlenül leíró tulajdonságvektorok és a neurális hálózatok projekciós rétegében kialakuló prediktív disztribúciós szóbeágyazások is.

2 Módszerek

2.1 A szemantikai keretek és keretelemek felismerésének feladata

Az 1. táblázatban felsorolt FrameNet keretek felismerését tanítottam be az erre a célra kidolgozott keretspecifikus neurális hálózatoknak.

Keret-azonosító	Keret (FR) neve	Keret gyakoriság szerinti sorszáma	Keret előfordulási gyakorisága	Keretelemek (FE) száma
73	*Leadership*	5.	499	13
173	*Buildings*	7.	420	12
408	*Manufacturing*	14.	277	13
191	*Natural_features*	16.	269	9
118	*Possession*	17.	260	7
990	*Capability*	18.	259	8
304	*People*	19.	257	8
34	*Discussion*	81.	87	12
1371	*Organization*	89.	79	8
141	*Certainty*	93.	74	7
172	*Commerce_sell*	95.	73	8
171	*Commerce_buy*	145.	50	9

1. táblázat. A kísérletben használt FrameNet keretek és keretelemek

A FrameNet 1.7-es változatának full-text kísérőkorpuszában 792 különböző szemantikai kerethez találtam példákat az őket felidéző 28783 szótokent annotáló címke formájában. Ezen annotációk körülbelül 9%-át használtam fel az itt bemutatott kísérletekben. A full-text kísérőkorpusz nem tartalmaz példát minden FrameNet kerethez, továbbá a szemléltetett kereteknek 51%-ához csupán 1-10 példát, további 15%-ához pedig 11-20 példát tartalmaz, ráadásul a hozzájuk tartozó ritkább keretelemek néha egyáltalán nem szerepelnek a példák közt. Mind a betanításhoz, mind a teszteléshez szükségesek voltak ilyen adatok, és a tesztelésnél csak olyan mondatokra támaszkodhattunk, amelyeket a betanítás során nem használt fel a rendszer. Összességében adathiány (data sparsity) miatt a keretek jelentős részét a folyamatból eleve kizártam. A kiválasztott tizenkét keret közül kettő nagyon gyakori volt a korpuszban (420-499 előfordulással), a további keretek közepes- és alacsony frekvenciájúak voltak. A *Commerce_buy* keret 50 előfordulása például (9 keretelem mellett) nagyon kevés

tanító- és tesztadatot eredményezett, azonban későbbi kvalitatív vizsgálatokban (a *Commerce_sell* kerettel együtt) érdekes adatokat szolgáltathat.

A keretelemek felismerése (néhány keretelem ritkasága miatt is) jóval nehezebb feladat volt. A keretelemtípusok keretenkénti száma az 1. táblázatban látható. Néhány példa a keretelemekre:

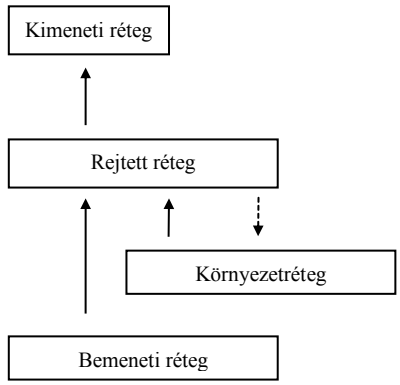
- A *Leadership* keretben: *Leader* („vezető”), *Role* („szerep”), *Governed* („irányított”), stb.
- A *Buildings* keretben: *Name* („név”), *Type* („típus”), *Possessor* („tulajdonos”), stb.
- *Commerce_buy* keretben: *Goods* („áru”), *Buyer* („vevő”), *Seller* („eladó”), stb.

A full-text kísérőkorpuszból vett példamondatok felét betanításra, a másik felét tesztelésre használtam, majd ezek szerepét felcseréltem és az elért eredményeket átlagoltam. Ez a keresztvalidációs eljárás – azzal együtt is, hogy csak kétszeres keresztvalidációhoz állt rendelkezésre elég erőforrás – a tesztelés fontos része volt, mivel a tanító- és a tesztadatok 50-50%-os elosztását kizárólag a keretek vonatkozásában sikerült elérni, a keretelemek esetében nem. Szótövesítést, alaktani vagy mondattani elemzést nem végeztem (ill. ilyen adatot nem használtam fel a korpuszból). Többszavas kifejezéseket, összetevőket nem kezeltem együtt, a szemantikai feladatot végrehajtó hálózattól vártam az ezeket alkotó szavak megfelelő (azonos) címkével történő annotálását. A többszavas kifejezések együttes kezelése és a mondattani összetevők előzetes azonosítása a rendszer teljesítményét minden bizonnyal növelné. Neurolingvisztikai megfigyelések által is motivált az ilyen irányú későbbi továbbfejlesztés, hiszen a nyelvfeldolgozás során az ELAN fázisban jól dokumentált módon lezajlik a „lokális mondattani jelenségek”, pl. bizonyos összetevők elemzése [6]. Egy ezzel analóg feldolgozási mozzanat az egyes keretelemekhez tartozó szócsoportok kiemelését, előfeszítését végezhetné el, egyúttal a többszavas kifejezések kezelését is elősegítené.

Mindegyik szemantikai keretet egy külön neurális hálózat ismert fel, mely Elman-elrendezésben [5] működött. A hálózattípusnak alkalmasnak kellet lennie időbeli mintázatok (szavak szekvenciájának) közvetlen megfigyelésére, ezért visszacsatolt („recurrent”) hálózattípust választottam. Elman a saját hálózati topológiáját „egyszerű visszacsatolt hálózatnak” („simple recurrent network”, SRN) nevezte el, ami a hatékony és gyors betaníthatóságra és alkalmazhatóságra is utal. Az Elman SRN az egyik legelső visszacsatolt hálózati topológia volt, és a nyelvészek számára azért is figyelemre méltó, mert a szerző a hálózattípus eredeti bemutatásakor is kiemelte és demonstrálta a konstrukció felhasználhatóságát nyelvi jelenségek felismerésében is [5]. Amennyiben a rendszer teljesítményének maximalizálása lett volna a cél, akkor összetettebb visszacsatolt hálózattípusok implementálásával (pl. LSTM) valószínűleg további javulást lehetett volna elérni, azonban jelen esetben ez nem volt fontos szempont, hiszen elsősorban a különböző szóreprezentációk összehasonlítását tűztem ki célul.

Az Elman SRN egy rejtett- és egy környezetrétegeket tartalmaz az 1. ábrán szemléltetett módon. A rejtett réteg minden egyes neuronja pontosan 1 környezetneuronhoz van hozzákapcsolva rögzített súllyal. A környezetréteg neuronjai a rejtett réteg idegsejtjeihez kapcsolódnak (jelen implementációban teljes projekcióval, azaz minden neuron a következő réteg összes neuronjához) tanítható súlyokkal. Az itt bemutatott kísérletekben a környezetréteg segítségével mondaton belüli rövid-

távú memóriát alakítottam ki: ennek a rétegnek az idegsejtjeit aktiváció nélküli (0) állapotba hoztam minden mondat utolsó szava után. A bemeneti réteg neuronjaitól a rejtett réteg feldolgozóegységeihez, a rejtett réteg idegsejtjeitől pedig a kimeneti réteg neuronjaihoz az információt teljes projekcióval, tanítható súlyokkal vezettem tovább.



1. ábra: Az Elman SRN topológia áttekintése

A bemeneti rétegen a mondat aktuális szavát jelenítettem meg 9 különböző módon kódolva (ld. 2.2. szakasz) külön kísérletsorokban. A bemeneti réteg mérete a használt szóreprezentációs módszer függvényében 300-10000 neuron volt.

Ahhoz, hogy a hálózatok megfelelő általánosítási képességgel rendelkezzenek, a rejtett rétegnek és az ahhoz közvetlenül kapcsolódó környezetrétegnek a méretét megfelelően alacsonyra kellett beállítani. Ez ebben az esetben 25 idegsejtet jelentett, amit kísérletezéssel választottam ki a 10-300 tartományból.

A kimeneti réteg mesterséges idegsejtjei közül 1 végezte el a szemantikai keretet felidéző lexikális egység („frame-evoking lexical unit”) címkézését, ezáltal a szemantikai keret felismerését. A keretelemeket egy kimeneti mintázat azonosította, melyben egy-egy neuron volt felelős egy-egy keretelem azonosításáért, valamint egy másik neuron jelezte, hogy a hálózat kimenete érvényes mintázatot tartalmaz. A kimeneteken a FrameNet full-text korpusz megfelelő keret- és keretelem kategóriáit reprezentáló mintázatok megjelenését vártam.

A hibát a tanítás során az SRBPTT („simple recurrent backpropagation through time”) algoritmus felhasználásával lépésenként csökkentettem. Egy-egy hálózat betanítását 1200 tanítási menetben végeztem el (a hibadiagramok alapján 800-1200 menet után stabilizálódott a kimeneti hiba a hibaminimum közelében a szemantikai keret és a szóábrázolási módszer függvényében), mindegyik menet egy teljes mondat betanításának felelt meg. A mondatok általagos hossza 21 szó volt (ezek voltak a betanítási menet eseményei; a súlyok nem az események, hanem a menetek végén változtak). A tanítási hibát sikerült a várt módon menedzselni, a feladat a kiválasztott eszközzel megoldható volt, a hálózat hatékonyan megjegyezte a tanítóban lévő FrameNet címkéket. Az új, korábban nem látott mondatokat (és számos először látott szót) tartalmazó tesztadatokra kapott pontossági és fedési értékeket a 3. szakaszban ismertetem.

2.2 Szavak ábrázolása a szemantikai feladatot végrehajtó hálózatok bemenetén

A szavakat a neurális hálózat számára numerikus adatokká kell alakítanunk, ennek 9 módszerét próbáltam ki, köztük olyan eljárásokat, amelyek a szavak disztribúciós tulajdonságait (nagy mintán megfigyelt együttes előfordulási adatait) kódolta.

1. 1HOT (one-hot, 1-az-N-ből): A szakirodalomban elterjedt megoldás, melyben a bemeneti vektor elemei közül egyet 1-re, a többit 0-ra állítjuk, és minden szótípushoz más vektort rendelünk. Új szótípus hozzáadása új elem bevezetésével történik, amelyet a neurális hálózatos kísérletekben a következő idegsejt-réteghez megfelelően hozzá kell kapcsolni, majd a hálózatot újratanítani.
2. COUNT-LOGFREQ: A disztribúciós szemantika [11] hagyományos gyakorlatának megfelelően minden szóhoz előállítottam egy olyan tulajdonságvektort, ami megmutatta, hogy az adott célszó más szavakkal milyen gyakorisággal fordult elő együtt egy nagy méretű (de szemantikai annotáció nélküli) korpuszban. Az ilyen eljárás során, például, ha az *iszik* célszót jellemezzük a *vizet*, *teát*, *kólából* és *haza* környezetszavakkal, akkor a *vizet*, *teát* és *kólából* környezetszavaknak megfelelő vektorelemek értéke magasabb lesz, a *haza* szóhoz tartozó vektorelem értéke alacsonyabb. A kapott vektorok egy valós vektortérben úgy kijelölik az adott szó helyét, hogy a hasonlóbb disztribúciójú szavakhoz tartozó vektorok hajlásszöge kisebb lesz (további részletekért ld. [11]). A disztribúció hasonlósága szemantikai, szintaktikai és morfológiai okoknak (együttesen és egymástól elválaszthatatlanul) köszönhető. A disztribúciós adatok összegyűjtéséhez a TC Wikipedia korpusz (http://nlp.cs.nyu.edu/wikipedia-data) véletlenszerűen kiválasztott 100 millió szavas részkorpuszát elemeztem. A többmilliárd szavas korpuszok potenciálisan jobb eredményt adnak, ugyanakkor az emberi tapasztalás korlátait messze túllépik. A TC Wikipedia korpuszból semmilyen annotációt nem használtam fel, és a célszavak 3+3 szavas környezetét vizsgáltam. Környezetszókként az 5000 leggyakoribb angol szót kerestem, ennyi lett a kapott tulajdonságvektorok elemeinek száma. Mivel néhány együttes előfordulásból (különösen a funkciószavak esetében) nagyon sokat találhatunk, a vektor elemeit súlyozni szükséges. A COUNT-LOGFREQ reprezentációban az együttes előfordulási gyakoriság logaritmusával számoltam.
3. COUNT-PPMI: A szóvektorokat a COUNT-LOGFREQ reprezentációnál ismertetett eljárással állítottam össze azzal a különbséggel, hogy a vektor elemeinek súlyozását másképpen végeztem: a célszavak és környezetszavak egyedi kölcsönös információját (EKI [7]; angolul: pointwise mutual information, PMI) használtam, amennyiben az pozitív érték volt, ellenkező esetben nulla lett a vektorelem értéke. Amennyiben a célszó (c) és a környezetszó (k) előfordulása független egymástól, akkor az együttes előfordulás valószínűsége $P(c) \times P(k)$, ehhez viszonyítjuk c és k megfigyelt együttes előfordulásainak számát ($M(c,k)$); a pozitív egyedi kölcsönös előfordulás értéke $pPMI=max(0,log(M(c,k) / P(c) \times P(k)))$.
4. RND-PPMI: COUNT-PPMI módszerrel készített szóvektorok mindegyikét egy másik, véletlenszám-generátorral kiválasztott szóvektorral felcseréltem, ezáltal a szóvektorokat megfosztottam valós disztribúciós (szövegkörnyezetet kódoló) tartalmuktól. A COUNT-PPMI szóábrázolással összehasonlítva az RND-PPMI reprezentáció alkalmas a disztribúciós információ hatásának megfigyelésére.

5. PRED-CBOW: Visszacsatolás nélküli, 1 rejtett réteget tartalmazó neurális hálózatban Mikolov és mtsai módszerével [8] létrehozott prediktív disztribúciós szóábrázolás. A CBOW („continuous bag of words") eljárás használata esetén a hálózat a bemeneti rétegen egy környezetablak szavait kapja, betanítás után a kimeneten pedig az ablak közepén álló szót jósolja meg. Számunkra nem a hálózat kimenete (a jóslás minősége), hanem a feladat megoldása során az adott célszó előállításához szükséges belső mintázat (a rejtett réteg aktivációs adatsora) az érdekes: ez a beágyazott mintázat lesz az adott célszó PRED-CBOW ábrázolása. A szakirodalomban kialakult gyakorlat szerint az így kapott aktivációs értékeket egy-egy vektor elemeinek tekintjük, a vektorok pedig minden szóhoz kijelölnek egy pontot egy sokdimenziós valós vektortérben úgy, hogy a disztribúciós szempontból hasonlóbb szavakhoz tartozó vektorok hajlásszöge kisebb lesz. A vektorok létrehozásához a TC Wikipedia fent említett részkorpuszát és a word2vec eszközt [9] használtam, 3+3 szavas környezetablak vizsgálatával. A vektorok 300 elemből álltak.
6. PRED-SKIPGRAM: a PRED-CBOW vektorokhoz hasonló eljárással létrehozott prediktív vektorok [8]. A skip-gram szóbeágyazás előállítására használt hálózat a bemenetén a célszót kapja meg, a kimeneten pedig a szó környezetét kell megjósolnia. Itt is igaz, hogy nem a jóslás pontossága, hanem a feladat megoldása során kialakuló aktivációs mintázatok (a rejtett réteg aktivációs szintjei) az érdekesek számunkra, ezeket a célszó disztribúciós tulajdonságait tömörítő szóbeágyazásokként kezeljük. Mérete: 300 valós érték minden szóvektorban.
7. RND-SKIPGRAM: PRED-SKIPGRAM módszerrel készített szóbeágyazások véletlenszám-generátorral kiválasztott párjait felcseréltem, ezt az eljárást minden szóra megismételtem. Az így kapott mintázatokban az adott célszó vonatkozásában valós disztribúciós adat nem maradt. Ez az ábrázolás a PRED-SKIPGRAM szóábrázolással összehasonlítva alkalmas a disztribúciós információ hatásának mérésére. Az RND-SKIPGRAM ábrázolást annak a megfigyelésnek az apropóján vezettem be, hogy a word2vec által generált PRED vektorokban nagyon sok a nullához közeli elem, ami befolyásolhatja a betanítás sikerét, amennyiben ezeket a vektorokat a további feldolgozás során is neurális hálózatokban használjuk fel. (A PRED-CBOW és PRED-SKIPGRAM ábrázolások ebből a szempontból hasonlóak, ezért RND-CBOW ábrázolást nem készítettem).
8. 1HOT + COUNT-PPMI: kombinált 1HOT és COUNT-PPMI reprezentáció minden célszóhoz. Mivel a disztribúciós ábrázolásmód esetén a szó és reprezentációja közt kölcsönösen egyértelmű megfeleltetés nem garantált, ugyanakkor a 1HOT reprezentációt éppen erre vezették be, így a kombinálásukból előny származhat. Az így készült reprezentációk nagy méretűek, esetünkben 5000 vektorelem a COUNT-PPMI reprezentációból és 5000 elem a 1HOT reprezentációból.
9. 1HOT + PRED-SKIPGRAM: kombinált 1HOT és PRED-SKIPGRAM ábrázolás minden szóhoz. Kipróbálásának oka a kölcsönösen egyértelmű megfeleltetés létrehozása prediktív szóbeágyazás mellett. Mérete 5000 + 300 elem vektoronként.

A prediktív szóábrázolások érdekes (bár ritkán tárgyalt) tulajdonsága, hogy úgy hozzuk őket létre, hogy egy nyelvészeti szempontból kevésbé dokumentált feladatot, a szókörnyezet nyelvi kategóriáktól független előrejelzését, illetve a környezet alapján a célszó előrejelzését tanulja meg egy neurális hálózat. A sikeres emberi kommunikáció szempontjából ez egy releváns feladat, hiszen a zajos környezetben az eredeti jel

felismerését nagymértékben segíti a megfelelő szavak előfeszítése, egyúttal hozzájárul ahhoz, hogy a jelek feldolgozása gyorsan és félreértésektől mentesebben történhessen meg.

2.3 A szoftverkörnyezet

A kísérletsor szoftveres infrastruktúráját a szerző hozta létre az alábbiak szerint. A kísérlet előkészítő szakaszában a FrameNet keretekről és keretelemekről gyűjtöttem adatokat erre a célra létrehozott programmal. Az előkészületek részeként a korpusz összes szavának minden szóreprezentációs mintázatát elő kellett állítani, amit szintén saját programmal végeztem el a PRED-SKIPGRAM és PRED-CBOW szóábrázolások kivételével, amelyek létrehozásához a word2vec eszközt [9] használtam. Ezután következett az a többlépcsős műveletsor, melyet minden szemantikai kerethez (12 db) és minden szóreprezentációhoz (9 db) külön elvégeztem, a kétszeres keresztvalidáció miatt pedig mindezt kétszer hajtottam végre, azaz összesen 216 kísérletről közlök összesített, átlagolt adatokat. Az eljárás lépései minden esetben ezek voltak:

1. A FrameNethez mellékelt full-text korpusznak az adott szemantikai keretet tartalmazó mondatait a neurális hálózati szimulátor által felismert bemeneti fájlokká alakítottam saját programmal, eközben a szavakat a megfelelő szóvektorokkal helyettesítettem a tanító- és tesztkorpuszban, valamint elhelyeztem a FrameNet kategóriacímkéknek megfelelő elvárt kimeneti mintázatokat.
2. Létrehoztam az adott kerethez és az adott szóreprezentációs módhoz tartozó mesterséges neurális hálózatot a LENS hálózatszimulátorban [10] és betanítottam azt a tanító adathalmazzal.
3. Ugyanezt a hálózatot teszteltem a tesztadatokkal, a tesztszimuláció során lementett kimeneti mintázatokat pedig saját programmal kiértékeltem, összehasonlítva a korpuszban látott és a hálózat kimenetén kapott szemantika keret- és keretelemcímkéket. Kiszámítottam a fedési és pontossági értékeket.
4. Keresztvalidáció céljából a tanító- és tesztadatokat felcseréltem, majd a 2-3. lépéseket megismételtem.

Az Elman hálózat paramétereinek beállításához előzetesen további kísérleteket végeztem (ennek során választottam ki a rejtett réteg méretét), valamint további vizsgálatokat hajtottam végre a leszámlálásos (COUNT) szóreprezentációk paramétereinek beállítása során (pl. a szövegablak méretének meghatározása). A szoftverkörnyezet létrehozása, ellenőrzése során értelemszerűen rengeteg további tesztfuttatást is végeztem. A szoftverfejlesztés és a kísérletek végrehajtása a 2014–2018 időszakban történt.

3 Eredmények

A neurális hálózatok a maximális teljesítményüket leszámlálásos disztribúciós szóreprezentációval nyújtották 91% pontosság és 86% fedés mellett a szemantikai keretek (FR) felismerése során. A keret*elemek* (FE) felismerése nehezebb feladat volt a következő okok miatt: *a)* néhány keretelem esetében nagyon kevés tanítóadat állt rendelkezésre, szélsőséges esetként volt olyan keretelem is, amihez csak 1 tanító- és 1

tesztadat volt; *b)* míg a keretfelismerés általában 1 szó (a keretet előhívó lexikai egység, „frame-evoking lexical unit") megfelelő címkézését igényli, a keretelemek általában több szóból állnak, amelyeket a mostani rendszerben egyesével kell felcímkézni, az összetevők előzetes kijelölése nem megoldott.

A 2. táblázat a keretek és a keretelemek felismerésének százalékos pontosságát (*p*), fedési értékét (*r*) és ezek harmonikus átlagát (*F*-mértékét) mutatja a 9 vizsgált szóábrázolás mellett.

Szóábrázolás	*p (FR)*	*r (FR)*	*F (FR)*	*p (FE)*	*r (FE)*	*F (FE)*
1HOT	91,1	64,8	*75,7*	53,8	40,4	*46,1*
COUNT-LOGFREQ	91,2	**86,4**	***88,8***	55,9	42,7	*48,4*
COUNT-PPMI	**92,5**	84,6	*88,4*	56,4	46,9	*51,2*
RND-PPMI	89,8	75,9	*82,3*	54,2	42,6	*47,7*
PRED-CBOW	88,2	84,1	*86,1*	56,4	**49,5**	***52,7***
PRED-SKIPGRAM	89,4	83,3	*86,3*	57,2	48,8	*52,6*
RND-SKIPGRAM	81,1	67,8	*73,8*	49,6	40,1	*44,4*
1HOT + COUNT-PPMI	91,9	84,7	*88,2*	**57,7**	45,7	*51,0*
1HOT + PRED-SKIPGRAM	90,0	86,3	*88,1*	56,4	49,2	*52,6*

2. táblázat. A szóábrázolás hatása a keret (*FR*) és keretelem (*FE*) felismerésre

3.1 Disztribúciós információ nélküli eredmények

A 1HOT (one-hot, 1-az-N-ből) kódolási módszer egyszerűsége és elterjedtsége okán fontos viszonyítási alap a további eredmények értékelése szempontjából. Ezzel a szóábrázolással a keretfelismerés pontossága magas (91%), a fedés viszont alacsony (65%) volt. Ilyen ábrázolás esetén csupán 1 bemeneti egység szolgáltat információt a további feldolgozáshoz a kapcsolatain keresztül (jelen esetben 300 súlyozható kapcsolaton keresztül), a többi bemenetről a rejtett réteghez vezető kapcsolatok aktiváció hiányában nem tudnak a feldolgozáshoz hozzájárulni (ebben a kísérletben kb. 1,5 millió ilyen helyzetben lévő súlyozható kapcsolatról beszélünk). Kismértékű véletlen zaj hozzáadása ezt a problémát valamilyen mértékben orvosolhatja – ezt a lehetőséget ebben a kísérletben nem próbáltam ki, de két másik randomizált ábrázolásmódról közlök adatokat.

A 1HOT ábrázoláshoz hasonlóan a randomizált RND-PPMI és RND-SKIPGRAM szóvektorok szintén csak a szavak azonosítására voltak felhasználhatók a hálózat számára, hiszen az adott célszót jellemző disztribúciós adatokat nem találunk bennük. A 1HOT ábrázolás (melyben csupán 1 aktív elem van), az RND-PPMI (számos aktív neuronnal) és az RND-SKIPGRAM (számos aktív neuronnal, sok nullához közeli elemmel) módszerek egymástól jelentősen eltérően viselkedtek. Az RND-SKIPGRAM reprezentáció eredményei alacsonyak voltak, alulmúlták a 1HOT teljesítményét is, az RND-PPMI vektorok azonban meglepően jó eredményeket hoztak csupán azzal, hogy *nem akadályozták* a neurális hálózat betanítását, működését.

Ezen a ponton azt is meg kell jegyeznünk, hogy ugyan a fenti reprezentációk a szavak általános, nagy korpuszban megfigyelt disztribúciójáról nem tárolnak informá-

ciót, a szemantikai címkézést végző rendszer mégis hozzáfér a szövegkörnyezetre vonatkozó adatokhoz (még ha sokkal kisebb mennyiségben is) a FrameNet tanítómondatokból.

3.2 A disztribúciós adatok hatása

A 1HOT módszerhez képest a legjobb disztribúciós ábrázolásmód F-mértékben mért előnye a keretek felismerése közben 13,1 százalékpont, a keretelemek címkézése esetén pedig 6,6 százalékpont volt. A fedés értékét különösen látványos módon (21,6 százalékponttal) növelte a disztribúciós információk megfelelő használata. A COUNT-PPMI módszer használatával a valós szódisztribúciós adatoktól megfosztott RND-PPMI vektorokhoz képest 6,1 százalékpontos F-mérték növekedés következett be.

A 3. szakasz bevezetőjében ismertetett okokból a keretelem-felismerés nehezebb feladat volt, ami a pontossági és fedési értékekben is tükröződött, azonban a disztribúciós adatok jótékony hatása itt is jól látható volt. Az RND-PPMI 47,7%-os eredményéhez (F-mérték) képest a valós disztribúciós adatokat tartalmazó COUNT-PPMI vektorok 51,2%-os (F) eredménye kb. 7 százaléknyi (3,5 százalékpontos) növekedést jelent. A hagyományos 1HOT és az ebben a feladatban legjobb (PRED-CBOW) szóábrázolás közti különbség pedig 14% (6,6 százalékpont) volt.

Baroni, Dinu és Kruszewski munkája [2] bemutatja, hogy a prediktív szóbeágyazások jobban teljesítenek a szokásos benchmark feladatok széles spektrumán, mint a hagyományos leszámlálós eljárással készített szóvektorok. Ez a várakozás ebben a kísérletsorban nem igazolódott, a prediktív (PRED-SKIPGRAM és PRED-CBOW) és leszámlálós (COUNT-LOGFREQ és COUNT-PPMI) módszerek közül ebben az esetben nem tudunk győztest hirdetni. A keretfelismerés során a COUNT módszerek, a keretelem-felismerésben pedig a PRED szóábrázolások voltak valamivel jobbak, kis különbséggel.

Általában is megfigyelhetjük, hogy a különböző disztribúciós eszközök (COUNT-LOGREQ, COUNT-PPMI, PRED-CBOW és PRED-SKIPGRAM) teljesítményének szórása ebben a feladatban alacsony volt. Ennek valószínűsíthető oka az, hogy a szemantikai címkézést végző rendszer a szóreprezentációk adatain kívül is hozzáfér a szövegkörnyezettel kapcsolatos információkhoz a FrameNet tanítómondatokból, hiszen egyrészt a visszacsatolt hálózat a környezetrétegben tárolt információk segítségével emlékszik a mondat korábbi szavaira, másrészt az idegsejtek közti súlyozott kapcsolatok hosszú távú memóriaként működnek az egész hálózatban az összes tanítómondatra vonatkozóan. Ez fontos különbség a disztribúciós szemantikai benchmark kísérletekhez képest, ahol a szavak szövegkörnyezetére vonatkozó adatokat tipikusan csak a szóreprezentációkból nyerhetjük ki, és csupán ezen adatokkal végezhetünk további műveleteket. Amennyiben a szóreprezentációkból olyan adatok hiányoznak, amelyek a feladat végrehajtásához lényegesek lennének, a hiányukat más forrásból nem tudjuk célzottan pótolni, míg ebben a kísérletsorban ez lehetséges volt a szemantikai címkézést végző neurális hálózat számára.

3.3 Szóábrázolások kombinálása

Mivel a disztribúciós szóvektorokkal a szavak és ábrázolások kölcsönösen egyértelmű megfeleltetése nem biztosított, kipróbáltam a szóvektorok és a 1HOT ábrázolás együttes alkalmazását is a tanulórendszer bemenetén, ezzel egyszerre megvalósítva a disztribúció kódolását és a szavak egyértelmű azonosításának feladatát. A *keretfelismerés* esetében a 1HOT + PRED-SKIPGRAM mérhető javulást eredményezett a PRED-SKIPGRAM önálló alkalmazásához képest (ld. még a korábbi SKIPGRAM-os megfigyeléseinket a 3.1 szakaszban). A COUNT-PPMI vektorokhoz adott 1HOT minták ugyanakkor nem hoztak további teljesítményjavulást a feladat megoldása szempontjából. A *keretelem-felismerési feladatban* a PRED-SKIPGRAM kiegészítése a 1HOT adatokkal a fedést ugyan enyhén növelte, de a pontosság csökkenése miatt az F-mérték változatlan maradt. A COUNT-PPMI vektorokhoz adott 1HOT minták pedig összességében még csökkentették is a pontosság és a fedés harmonikus átlagát.

4 Konklúzió

A FrameNet kiválasztott szemantikai kategóriáit tanuló és felismerő neurális hálózatot 9 különböző szóreprezentáció felhasználásával próbáltam ki egy komplex kísérletsorozatban. A bemutatott kísérletek alátámasztják, hogy a szemantikai keretek felismerésére lehetőség van egyszerű visszacsatolt hálózatokkal, és a feladat végrehajtását elősegíti a disztribúciós adatok megjelenítése a szóreprezentációkban. A továbbfejlesztés legfontosabb területe a keretelemek tekintetében a mondattani összetevők azonosítása és együttes címkézése lehet, ezen kívül további visszacsatolt neurális hálózattípusok kipróbálása és a vizsgált szemantikai keretek körének bővítése jelent közvetlen továbblépési lehetőséget.

Bibliográfia

1. Baker, C.F., Fillmore, C.J., Lowe, J.B.: The Berkeley FrameNet project. In: Proceedings of the COLING-ACL, Montreal (1998)
2. Baroni, M., Dinu, G., Kruszewski, G.: Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In: Proceedings of ACL (2014) 238–247
3. Bullinaria, J.A., Levy, J.P.: Extracting semantic representations from word co-occurrence statistics: A computational study. Behavior Research Methods 39 (2007) 510–526
4. Bullinaria, J.A. & Levy, J.P.: Extracting semantic representations from word co-occurrence statistics: stop-lists, stemming and SVD. Behavior Research Methods 44 (2012) 890–907
5. Elman, J.L.: Finding structure in time. Cognitive Science 14 (1990) 179–211
6. Friederici, A.D.: The Brain Basis of Language Processing: From Structure to Function. Physiological Reviews 91 (2011) 1357–1392
7. Kálmán L.: Már megint bakot lövünk. https://qubit.hu/2018/07/15/mar-megint-bakot-lovunk (2018)
8. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient Estimation of Word Representations in Vector Space. https://arxiv.org/abs/1301.3781. (2013)

9. Mikolov, T., Sutskever, I., Chen, K, Corrado, G.S. & Dean, J.: Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems (2013) 3111–3119
10. Rohde, D.L.T.: LENS: The light, efficient network simulator. Technical Report CMU-CS-99–164. Carnegie Mellon University, Department of Computer Science, Pittsburgh, PA (1999)
11. Tóth, Á.: The Company that Words Keep: Distributional Semantics. Debrecen University Press, Debrecen (2014)